23ISPC401

MANGALORE INSTITUTE OF TECHNOLOGY & ENGINEERING

(A Unit of Rajalaxmi Education Trust®, Mangalore)
Autonomous Institute affiliated to VTU, Belagavi, Approved by AICTE, New Delhi
Accredited by NAAC with A+ Grade & ISO 9001:2015 Certified Institution

Model Question Paper
Seventh Semester B.E Degree Examination

PARALLEL COMPUTING

Time: 3 Hours **Max. Marks: 100**

Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.
2. M: Marks, L: RBT (Revised Bloom's Taxonomy) level, C: Course outcomes.

		Module -1	M	L	C
Q1	a.	Define parallel computing. Explain the main motivations for using parallel computing in modern systems.	6	L2	CO1
	b.	Explain Flynn's classification of parallel computers. Illustrate with examples.	7	L2	CO1
	c.	What is the cache coherence problem in parallel systems? Describe the different methods used to maintain cache coherence.	7	L2	CO1
		OR			
Q2	a.	Distinguish between: i) Data parallelism and Task parallelism ii) Static threads and Dynamic threads	6	L2	CO1
	b.	What is an interconnect in a parallel computer system? Describe the different types of interconnects used in parallel architectures. Support your answer with neat diagrams.	7	L2	CO1
	c.	Discuss the means of coordinating processes/threads with an example.	7	L2	CO1
		Module- 2			
Q3	a.	State and explain Amdahl's law with an example. How does Amdahl's law still apply to GPU programs under certain conditions?	6	L2	CO2
	b.	What are the main challenges in handling input/output (I/O) in MIMD systems? State the rules that should be followed for safe I/O operations in parallel programs.	6	L2	CO2
	c.	Define speedup and efficiency in the context of parallel computing. A parallel program executes on 4 processors and gives an execution time of 25 seconds. On a single processor, the same program takes 80 seconds. Comment on the scalability of the program.	8	L3	CO2
		OR			
Q4	a.	What is scalability in parallel systems? Distinguish between strong scalability and weak scalability with examples.	6	L2	CO2
	b.	How does problem size affect speedup and efficiency in parallel programs? Illustrate with examples for small, medium, and large problem sizes.	6	L2	CO2
	c.	A multiprocessor system has 8 processors, each executing 2 Giga instructions/sec. If 75% of a program is parallelizable, calculate the speedup using Amdahl's Law. Also calculate the effective execution rate in Giga instructions/sec for the whole system.	8	L3	CO2

Module – 3					
Q5	a.	Explain how MPI_Send and MPI_Recv perform communication between two MPI processes, illustrating with a program that shows basic send–receive operations.	10	L2	CO3
	b.	Demonstrate the working of any parallel sorting algorithm. Compare the same with the serial sorting method.	10	L3	CO3
OR					
Q6	a.	Explain the working of MPI_Scatter and MPI_Gather along with their syntax, illustrating with an example program.	10	L2	CO3
	b.	Apply tree-structured communication to compute a global sum and compare it with linear communication using an example diagram.	10	L3	CO3
Module – 4					
Q7	a.	Explain the syntax and functionality of OpenMP critical, atomic, and lock constructs, illustrating each with a suitable example.	10	L2	CO4
	b.	Apply the reduction clause in OpenMP to explain its syntax and internal working and demonstrate how it performs a SUM operation using a suitable example.	10	L3	CO4
OR					
Q8	a.	Explain how OpenMP allows programmers to break a computation into tasks, illustrating with the structure of an OpenMP program that computes n Fibonacci numbers using tasks.	10	L2	CO4
	b.	Explain the scheduling mechanisms in OpenMP. Write an OpenMP program that divides the Iterations into chunks containing 2 iterations, respectively (OMP_SCHEDULE=static,2). Its input should be the number of iterations, and its output should be which iterations of a parallelized for loop are executed by which thread.	10	L3	CO4
Module – 5					
Q9	a.	Explain the concept of GPGPU and describe the key architectural differences between a GPU and a CPU that make GPUs suitable for data-parallel workloads.	10	L2	CO5
	b.	Demonstrate, with a suitable CUDA program, how vector addition is implemented on the GPU, including the kernel definition and the corresponding kernel launch.	10	L3	CO5
OR					
Q10	a.	Describe the CUDA execution model by explaining the roles of threads, thread blocks, grids, and warps in achieving massive parallelism.	10	L2	CO5
	b.	Illustrate the implementation of CUDA Trapezoidal Rule I, showing how threads compute partial sums and how atomicAdd is used to update the final result while avoiding race conditions.	10	L3	CO5

-*****-